FACULTY OF COMPUTING
& INFORMATION TECHNOLOGY
KING ABDULAZIZ UNIVERSITY

كلية الحاسبات
وتقنية المعلومات
جامعة الملك عبدالعزيز

FCIT Journal
KAU

Gym System

DB Design
CPCS-241 Database I – Fall 2022 – Project

Name:

Amal Mohammed Almaimouni

Content

PART I: Analysis

1 Problem Definition and Data Requirements

1.1 Problem Description

As awareness of health and sport incredibly increase many people intend to ask for a membership in the gym. Also, services and capability have implied progressive growth for athletes. The goal of this system to provide gyms by professional, flexible, and consistent database that maintain the ease and quality of the system. To fulfil members conformable and goals

1.2 Data Requirements

1. **Department**

   Each department has a name, ID, and address.

2. **Employee**

   Each employee has a name, SSN, ID, job title, age and phone number.

   - Trainer_Employee: each trainer has qualification and experience.

3. **Member**

   Each member has a name (first, middle and last), ID, SSN, sex, date of birth, address, phone number and e-mail.

4. **Classes**

   Each class has a name, unique class reference number, trainer and number of members.

5. **Class Type**

   Classes are categorized into types such as endurance, dancing and cardio. Each class type has ID, name and description.

6. **Machine and Equipment**

   Each machine and equipment have a name, unique ID. Such as stationary bike, leg press machine and vending machine.

7. **Discount**

   The gym offers discounts for special occasions, such as discounts for getting new membership and renewing current membership. Each discount has the offer name, discount amount, unique date and discount number.

8. **Membership**

   Each membership has a unique ID, duration, name and payment amount.

9. **In-Body Test**

   InBody test provides a detailed breakdown of member's weight in terms of muscle, fat, and water percentage. Each in-body test has a unique ID, member's height, member's weight, age, and date of test.

10. **Room**

    Each room has a room number and floor.

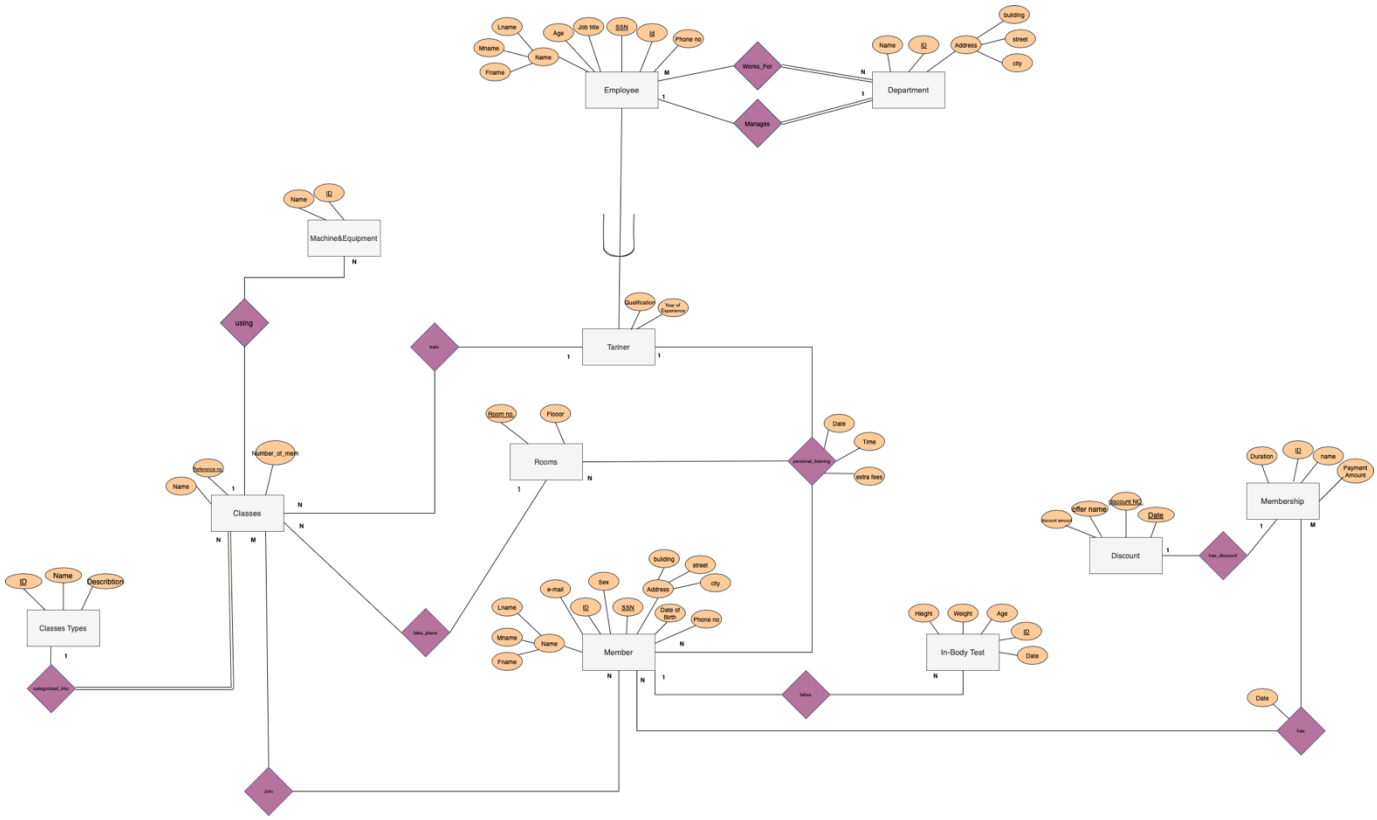
1.3 Business rules

- Each department must have exactly one manager.
- Each department must have employee working on it and each employee must belong to at least one department.
- Each member has a membership.
- Each class type can be assigned to many classes.
- Each member has availability to join any class and classes are conducted when there are several members enrolled.
- Each trainer can conduct classes, and not all trainers should conduct a class.
- Each trainer can work as a personal trainer for many members in any room.
- Each room can be available for a trainer to train many members.
- Each member has a trainer in any room.
- There are classes need to use equipment.
- Each membership can have a discount applied on it.
- Each member can have many In-body test.
- Each room can hold many classes, and each class take a place in one room.

PART II: DB Design

2 ER Diagram Design

2.1 ER Diagram



2.2 Design of Business Rules

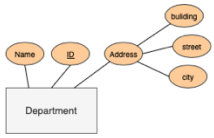
Business Rule	Design Decisions	Justification (if any)
Each department must have exactly one manager, and one manager can manage one department.	1:1 binary relationship.	Only one manager can manage one department.
Each department must have employee working within it and each employee must belong to at least one department.	N:M binary relationship	Each department has employees working within it, and employees must work for at least one department.
Each member has a membership.	N:M binary relationship	Each member can have only one membership.
Each class type can be assigned to many classes.	1:N binary relationship	The "class type" entity is an enumeration entity
Each member has availability to join any class and classes are conducted when there are several members enrolled.	N:M binary relationship	There will be classes when there are several members enrolled.
Each trainer can conduct classes, and not all trainers should conduct a class.	1:N binary relationship	A trainer can train many classes; however, a class can have one trainer.
Each trainer can work as a personal trainer for many members in any room.	1:N:N ternary relationship	Each trainer may work as a personal trainer; however, not all trainers must work as a personal trainer.
Each room can be available for a trainer to train many members.	1:N:N ternary relationship	Each room can be available for one trainer for number of members.
Each member has a trainer in any room.	1:N:N ternary relationship	Members can have a trainer in any room.
Each membership can have a discount applied on it.	1:1 binary relationship	-
There are classes need to use equipment.	1:N binary relationship	A class can use multiple equipment.
Each member can have many In-body test.	1:N binary relationship	Each In-body test can be taken by a member.
Each room can hold many classes, and each class take a place in one room.	1:N binary relationship	-

3 ER-to-Logical Schema Mapping

3.1 Mapping of Regular Entity Types

Department

ID	Name	Building	Street	City

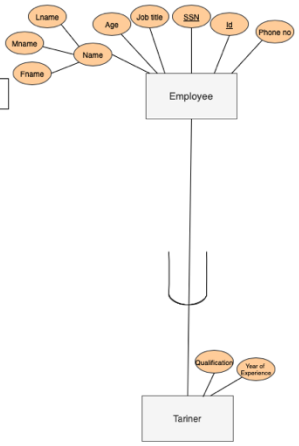


Employee

ID	SSN	Fname	Mname	Lname	Age	JobTitle	Phone no.

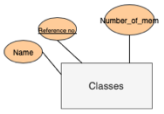
Trainer

Qualification	Year_of_Expreince	Emp_ID



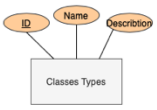
Classes

Reference no.	Name	Number_of_Mem



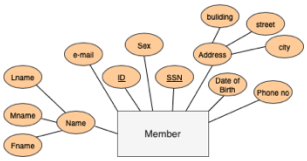
Class Types

ID	Name	Description



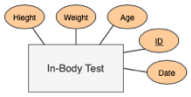
Members

SSN	ID	FName	MName	LName	e-mail	City	Street	Buliding	Sex	Date_of_Birth	Phone no.



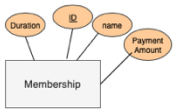
In-body Test

ID	Age	Hieght	Weight	Date



Membership

ID	Name	Duration	Payment amount

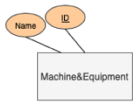


Discount

Discount_Number	Date	Offer_Name	Discount_Amount

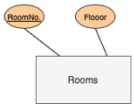
Machine and Equipment

ID	Name



Rooms

RoomNo.	Floor

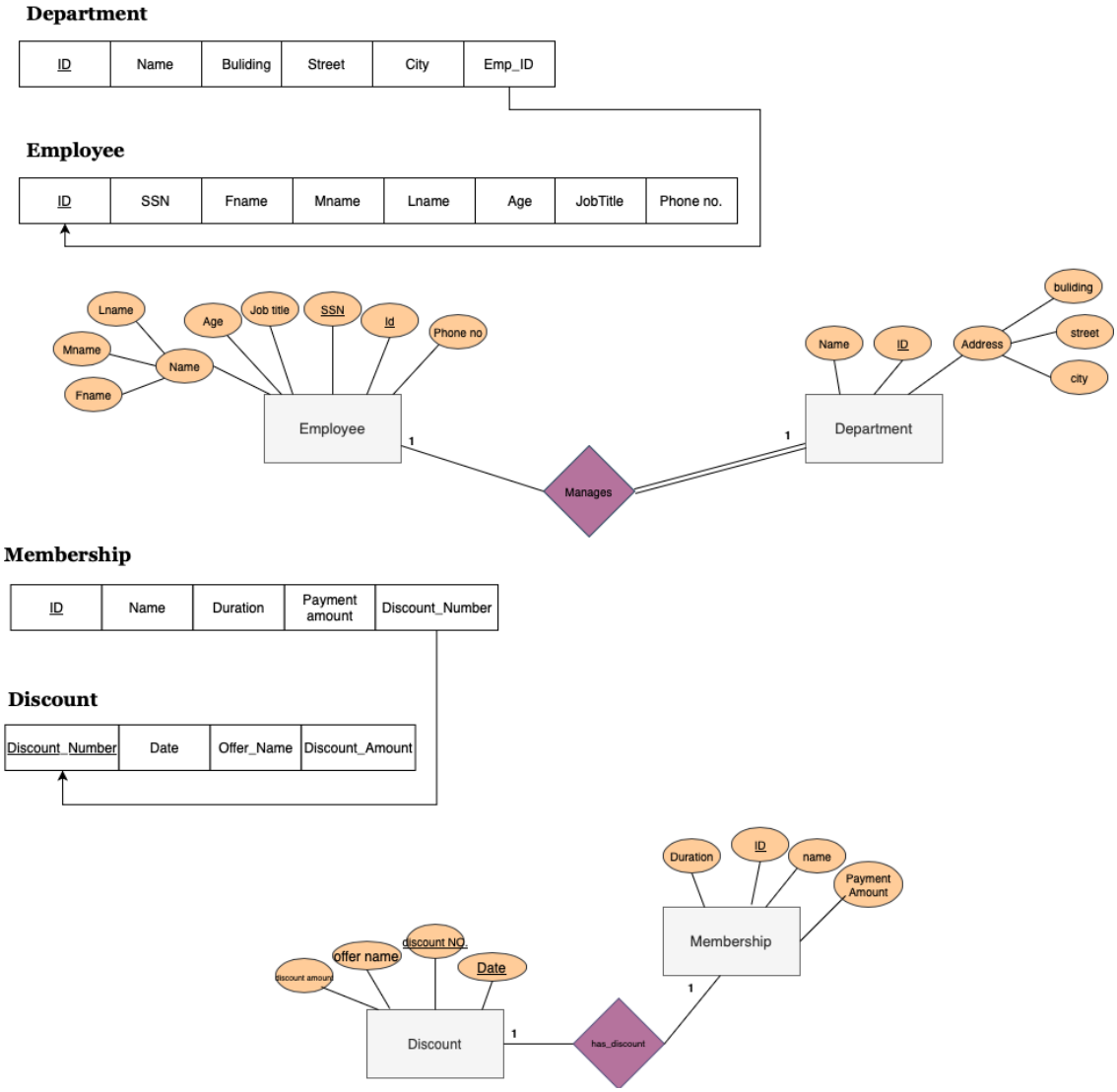


3.2 Mapping of Weak Entity

None.

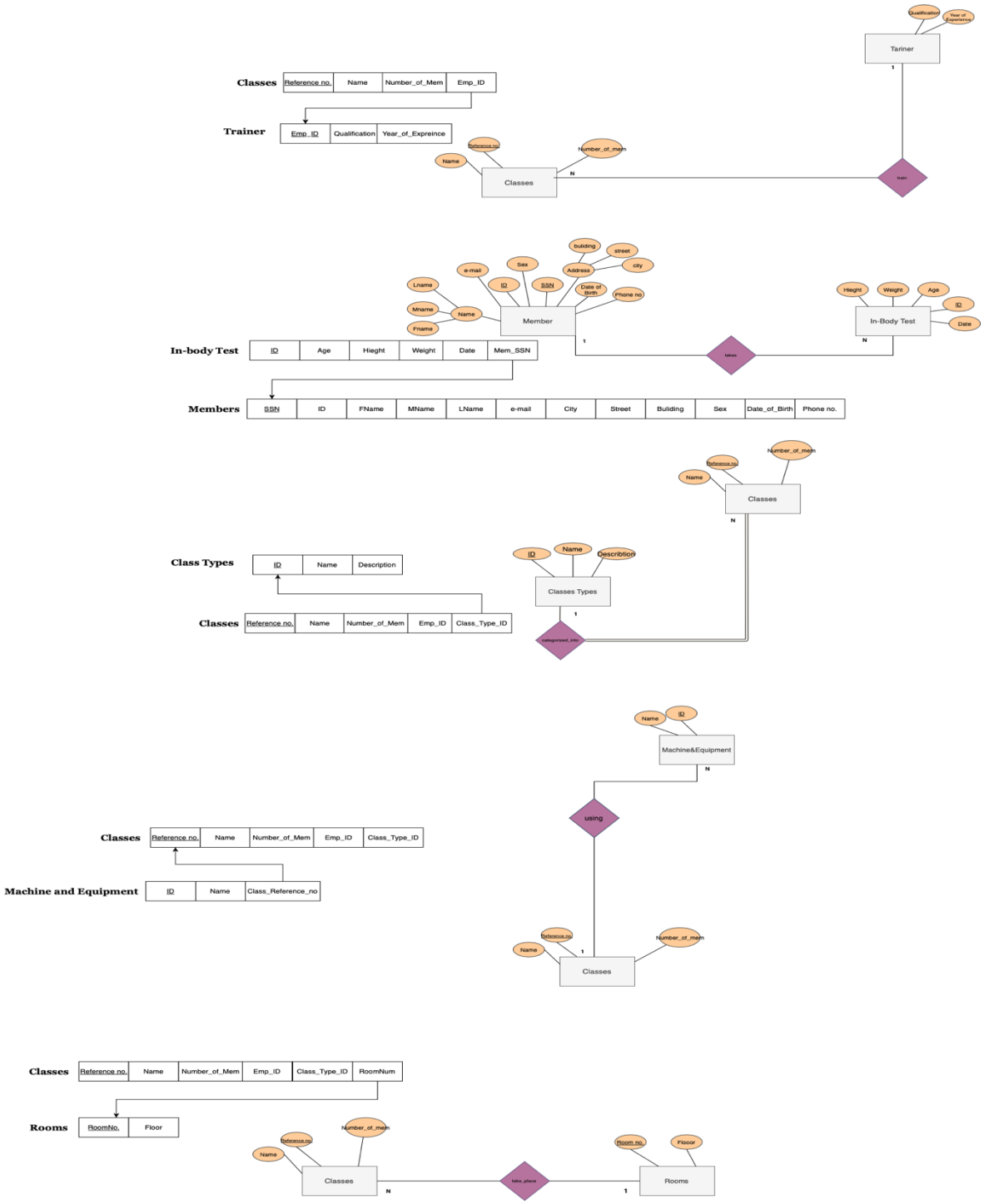
3.3 Mapping of 1-1 Binary Relatioship Type

When the relationship is limited only one element of each pary of the realtioship, where each element of the first party is related to only one element of the other, then the most approprite cardinality ration is one-to-one 1:1.



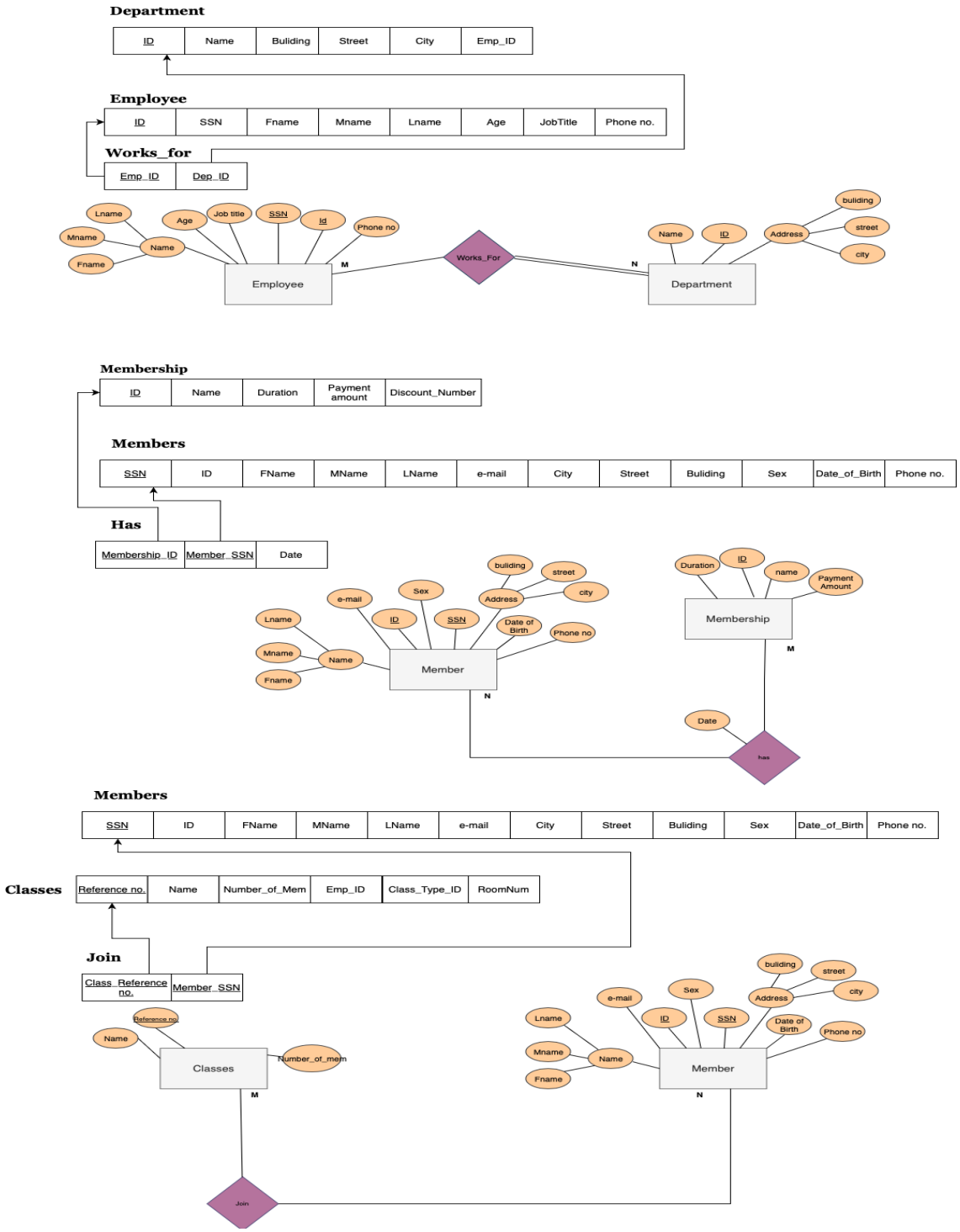
3.4 Mapping of 1-N Binary Relationship Types

When the relationship makes connection between one elemenet of the first party with sevral element of the second one but not vice versa, the most approprite cardinlity relation is one-to-many 1:N.



3.5 Mapping of Binary M-N Relationship Types

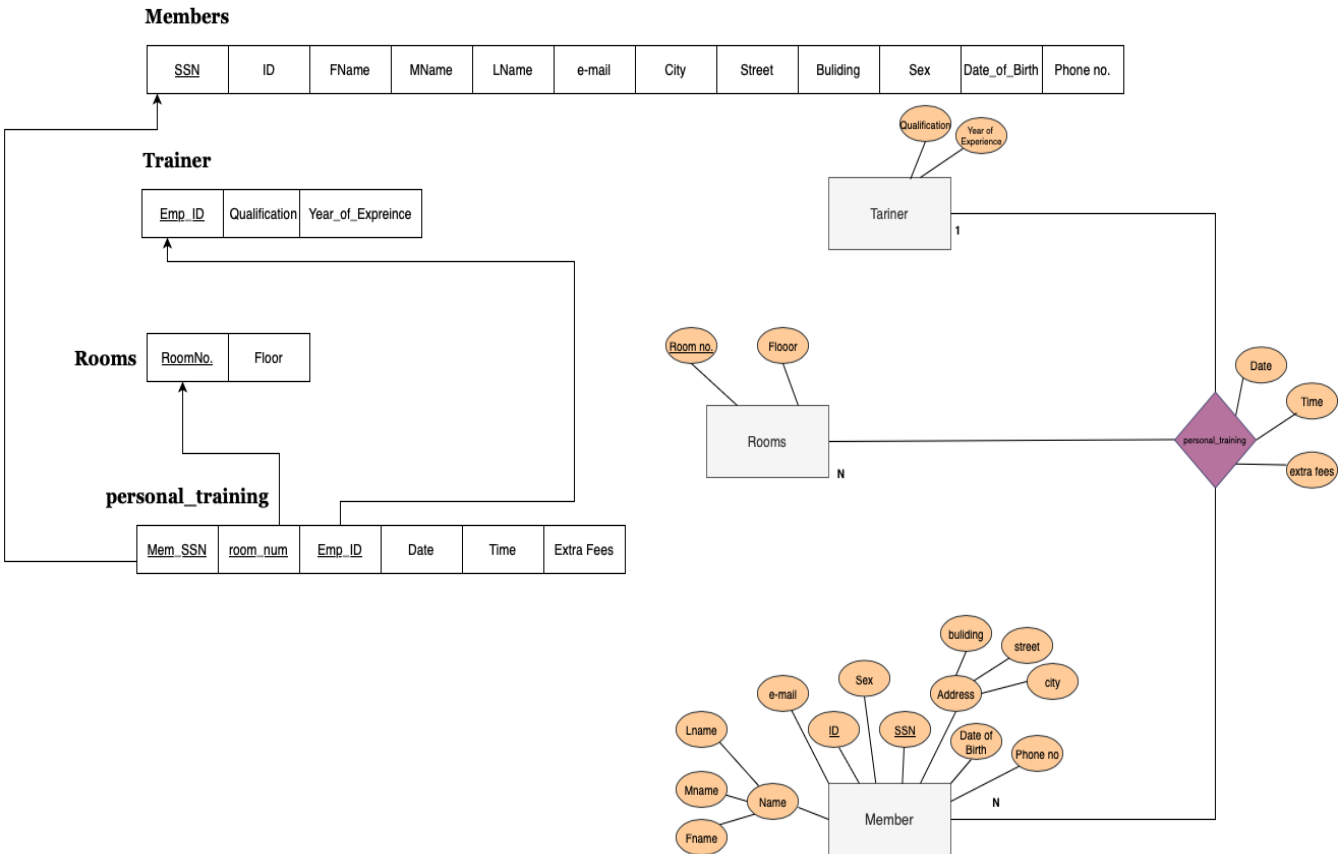
The relationship that links several elements of the first party with many elements of the second, since an element of the first party may be related to several elements of the second party and vice versa, this relationship is considered as many-to-many N:M.



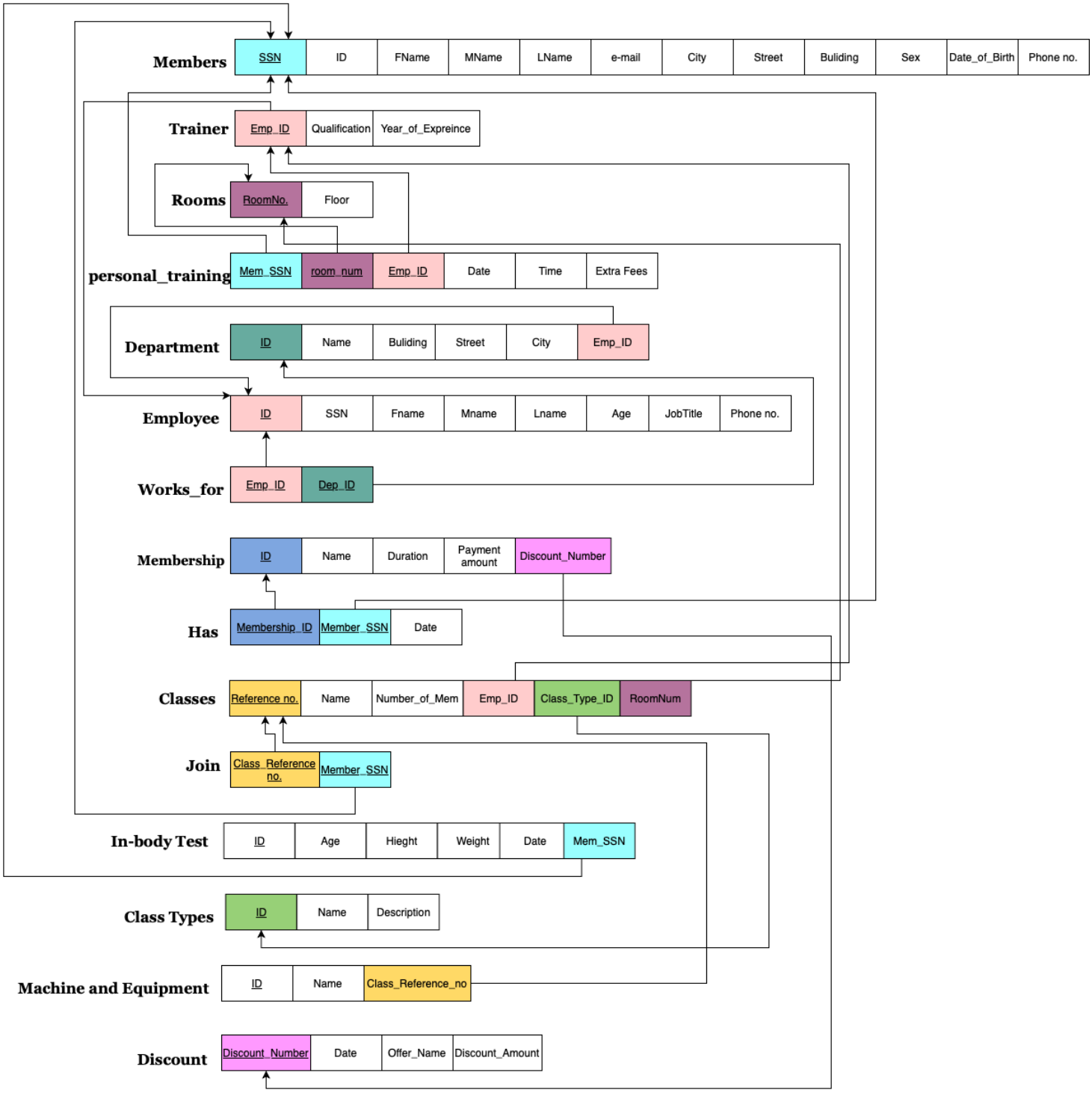
3.6 Mapping of Multivalued Attributes

None.

3.7 Mapping of N-ary Relationship Types



3.8 Schema Diagram



4 Normalization

4.1 First Normal Form

In the first normal form it is required to not have composite attributes, multivalued attributes and nested relations. We have 3 composite attributes in the Employee, Classes and Member entities. We solved them by adding a column for each composite attribute. Therefore, we achieved the 1NF.

Members	SSN	ID	FName	MName	LName	e-mail	City	Street	Buliding	Sex	Date_of_Birth	Phone no.

Trainer	Emp_ID	Qualification	Year_of_Expreince

Rooms	RoomNo.	Floor

Department	ID	Name	Buliding	Street	City	Emp_ID

Employee	ID	SSN	Fname	Mname	Lname	Age	JobTitle	Phone no.

Membership	ID	Name	Duration	Payment amount	Discount_Number

Classes	Reference no.	Name	Number_of_Mem	Emp_ID	Class_Type_ID	RoomNum

In-body Test	ID	Age	Hieght	Weight	Date	Mem_SSN

Class Types	ID	Name	Description

Machine and Equipment	ID	Name	Class_Reference_no

Discount	Discount_Number	Date	Offer_Name	Discount_Amount

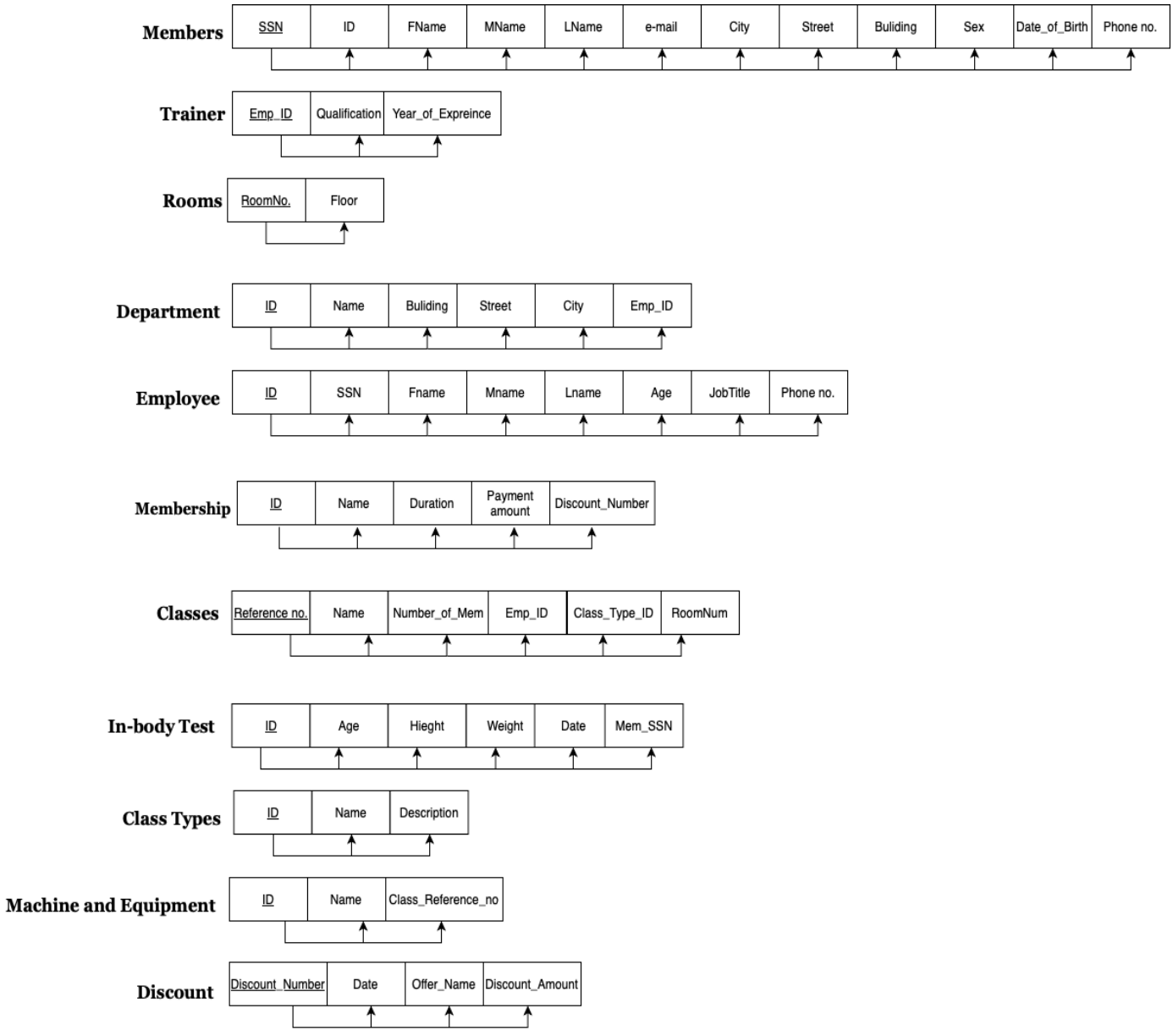
4.2 Second Normal Form

In the second normal form it is required for the schema to be in the 1NF and every non key attribute is fully functionally dependent on the prime attribute.

- **Department**

  ID is the only prime attribute of the Department relation, all non-prime attributes are fully functionally dependent on ID; therefore, no alteration is made.

- **The same goes on all the rest relations.**



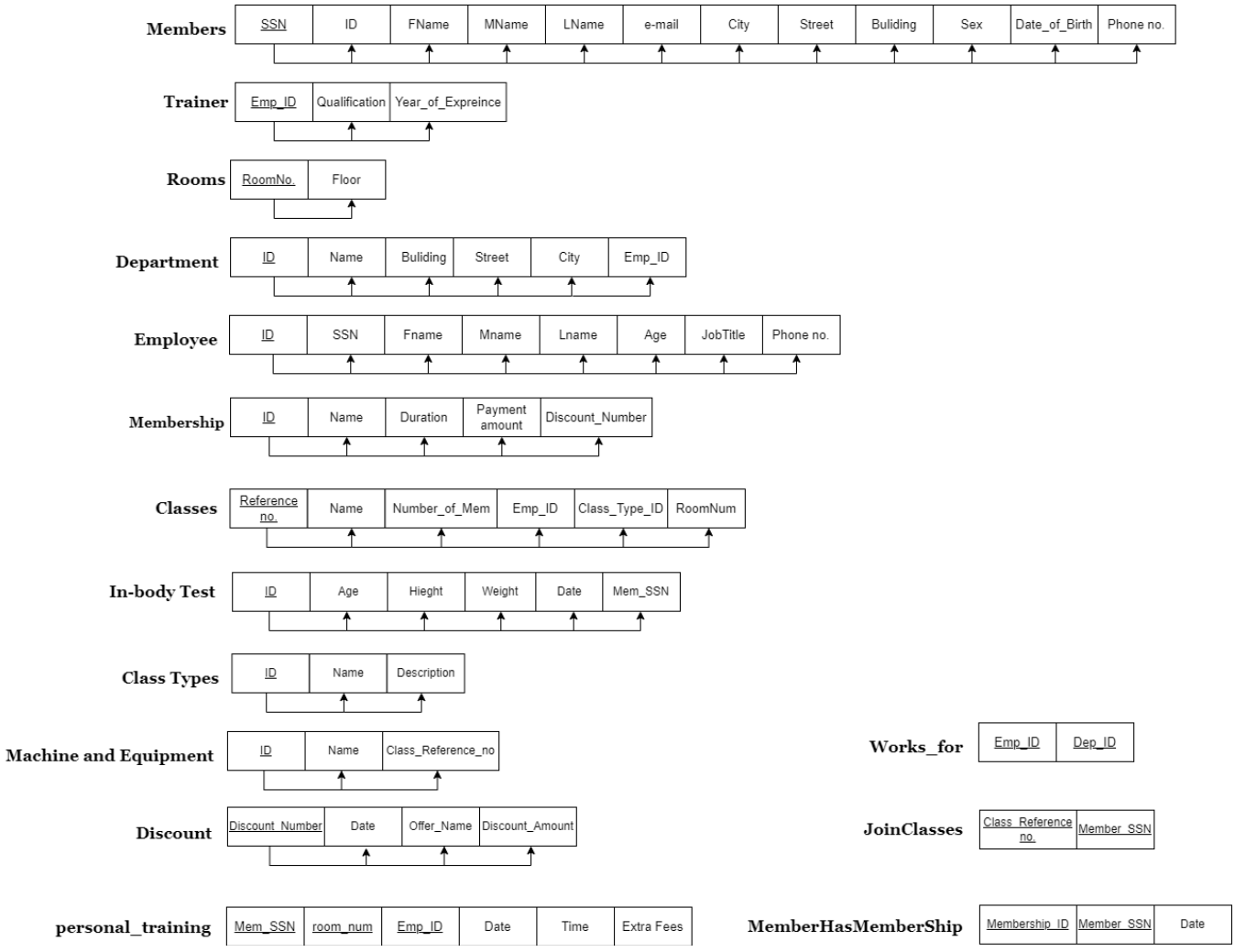
4.3 The Third Normal Form

In our schema all relations are in the 3NF; since the schema is in the 2NF and there is no transitive functional dependency from a non-prime attribute on the primary key.

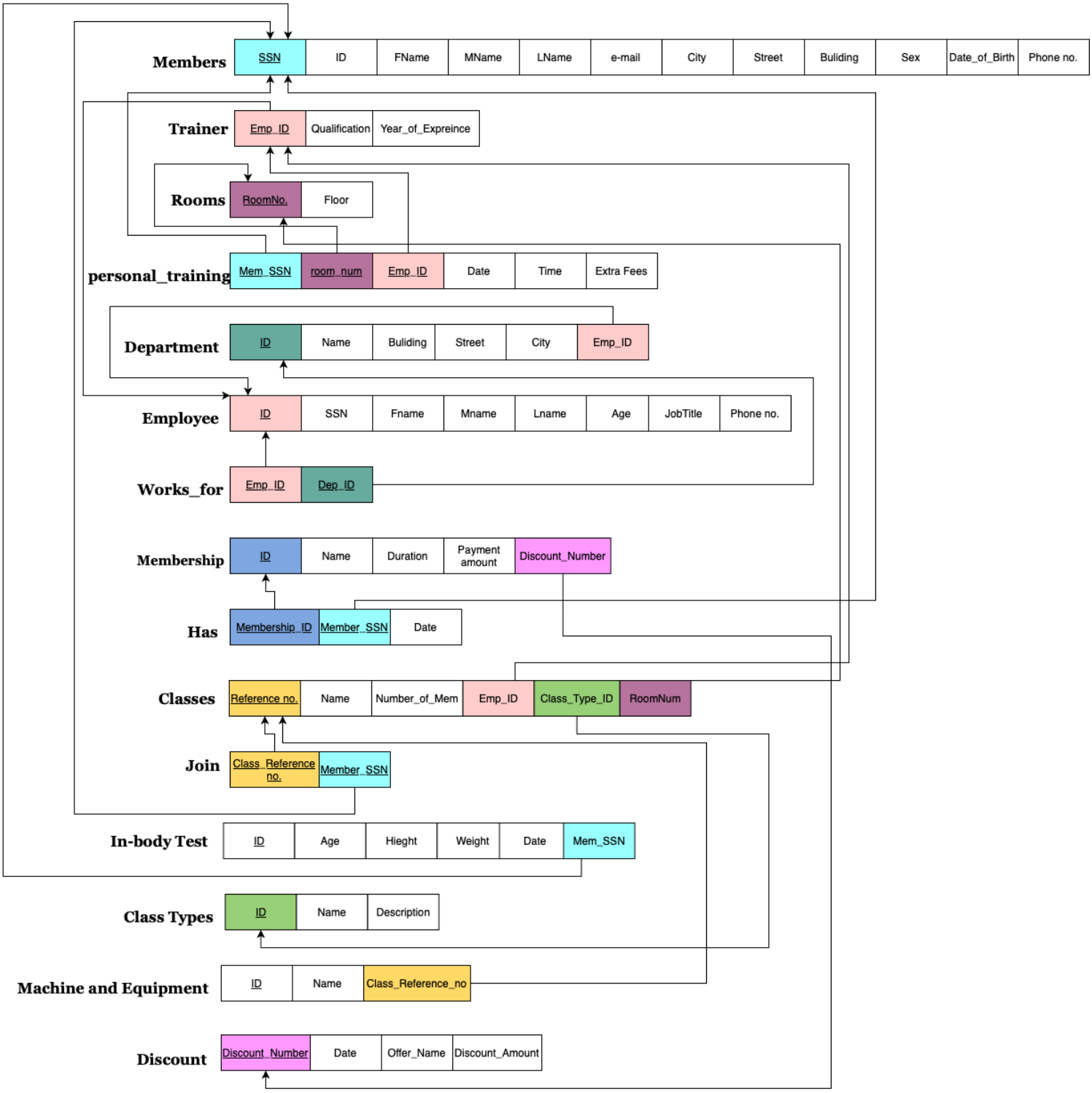
- **Department**

  Department relation is already in the 3NF, since all attributes depends on the primary key, and there are no transitive dependencies.

- **The same goes on all the rest relations.**



5 Final DB Schema Diagram



PART III: IMPLEMENTATION

6 Table Creation Script

6.1 <Members> TABLE

```
create table Members (
    ssn number(10) not null,
    ID VARCHAR2(30) unique not null,
    Fname varchar2 (30) not null,
    Mname varchar2(30),
    lname varchar2(30),
    email varchar2(30),
    city varchar2(30),
    street varchar2(30),
    building number(5),
    sex varchar2(10),
    birth_Date DATE,
    Phone_no number(10) unique,
    constraint members_pk primary key (ssn));
```

6.2 <Trainer> TABLE

```
create table Trainer (
    emp_Id number(10) not null,
    qualification VARCHAR2(60),
    Year_of_Experience number (5),
constraint trainer_pk primary key (emp_Id) )
;

alter table Trainer
    Add constraint traineremp_Id_fk foreign key(emp_Id) references employee(Id) on
delete cascade
;
```

6.3 <InBody_Test> TABLE

```
create table InBody_Test (
    Id number(10) not null,
    age number(5),
    height float,
    weight float,
    test_Date date,
    mem_SSN number(10),
constraint inBodyTest_pk primary key (Id) )
    ;
alter table InBody_Test
Add constraint InBody_Test_fk foreign key(mem_SSN) references Members(ssn) on delete
cascade
    ;
```

6.4 < department > TABLE

```
create table department (
    Id number(10) not null,
    Name varchar2(20) not null,
    Building number(5),
    Street varchar2(20),
    city varchar2(20),
    emp_Id number(10) not null,
constraint department_pk primary key (Id) )
;

alter table department
Add constraint departmentEmpID_fk foreign key(emp_ID) references employee(Id) on
delete cascade
    ;
```

6.5 < employee > TABLE

```
create table employee (
    Id number(10) not null,
    ssn number(10) not null,
    fname varchar2(20),
    mname varchar2(20),
    lname varchar2(20),
    age number(3),
    jop_Title varchar2(20),
    phone_No number(10) unique,
constraint employee_pk primary key (Id) )
;
```

6.6 < Membership> TABLE

```
create table Membership (
    Id number(10) not null,
    Name varchar2(20),
    Duration number(5),
    payment_Amount number(5),
    discount_No Number(10),
constraint Membership_pk primary key (Id) )
;

alter table Membership
Add constraint membershipDiscountNo_fk foreign key(discount_No) references
discount(discount_No) on delete cascade
;
```

6.7 <classes> TABLE

```
create table classes (
    reference_No number(10) not null,
    Name varchar2(25),
    classesTypeId number(10) not null,
    Room_No number(5),
    members_Number number(5),
    emp_Id number(10) not null,
constraint classes_pk primary key (reference_No) )
;

alter table classes
Add constraint classesEmpID_fk foreign key(emp_Id) references Trainer(emp_Id) on
delete cascade
;

alter table classes
Add constraint classesRoomNo_fk foreign key(Room_No) references rooms(roomNo) on
delete cascade
;
```

6.8 <Machine_And_Equipment> TABLE

```
create table Machine_And_Equipment (
    Id number(10) not null,
    Name varchar2(20),
    Class_Reference_No number(10) not null,
constraint Machine_pk primary key (Id) )
    ;
alter table Machine_And_Equipment
Add constraint machineClassRefNo_fk foreign key(class_Reference_No) references
classes(reference_No) on delete cascade
;
```

6.9 <class_Type> TABLE

```
create table Machine_And_Equipment (
    Id number(10) not null,
    Name varchar2(20),
    Class_Reference_No number(10) not null,
constraint Machine_pk primary key (Id) )
;
```

6.10 <discount> TABLE

```
create table discount (
    discount_No number(10) not null,
    discount_Date date,
    offer_Name varchar2(20),
    discount_Amount Number(5),
constraint discount_pk primary key (discount_No))
    ;
```

6.11 <rooms> TABLE

```
create table Rooms (
    roomNo number(5) not null,
    floor VARCHAR2(5),
constraint roomNo_pk primary key (roomNo) )
   ;
```

6.12 <personal_Training > TABLE

```
create table personal_Training (
    Mem_ssn number(10) not null,
    roomNo number(5) not null,
    emp_ID number(10) not null,
    training_Date varchar2(25),
    training_Time varchar(10),
    Extra_Fees number(5),
constraint training_pk primary key (Mem_ssn, roomNo, emp_ID) )
;

alter table personal_Training
Add constraint trainingMemSSN_fk foreign key(Mem_ssn) references Members(ssn) on
delete cascade
;


alter table personal_Training
Add constraint trainingRoomNo_fk foreign key(roomNo) references Rooms(roomNo) on
delete cascade
;


alter table personal_Training
Add constraint trainingEmpID_fk foreign key(emp_ID) references Trainer(emp_Id) on
delete cascade
;
```

6.13 < works_for > TABLE

```
create table works_for (
    emp_Id number(10) not null,
    dep_Id number(10) not null,
constraint worksFor_pk primary key (emp_Id, dep_Id) )
   ;

alter table works_for
Add constraint worksForEmpID_fk foreign key(emp_ID) references employee(Id) on delete
cascade
;

alter table works_for
Add constraint worksForDepID_fk foreign key(dep_ID) references department(Id) on
delete cascade
;
```

6.14 < MemberHasMemberShip > TABLE

```
create table MemberHasMemberShip (
    membership_Id number(10) not null,
    member_ssn number(10) not null,
    membership_Date date,
constraint has_pk primary key (membership_Id, member_ssn) )
;

alter table MemberHasMemberShip
Add constraint hasMembershipID_fk foreign key(membership_Id) references
Membership(id) on delete cascade
;


alter table MemberHasMemberShip
Add constraint hasMemberSSN_fk foreign key(member_SSN) references Members(ssn) on
delete cascade
;
```

6.15 <joinClass> TABLE

```
create table joinClass (
    class_Reference_No number(10) not null,
    member_SSN number(10) not null,
constraint join_pk primary key (class_Reference_No, member_SSN) )
;

alter table joinClass
Add constraint joinClassRefNo_fk foreign key(class_Reference_No) references
classes(reference_No) on delete cascade
;


alter table joinClass
Add constraint joinMemberSSN_fk foreign key(member_SSN) references Members (ssn) on
delete cascade
;
```

7 Constraints Script

In this subsection, the business rules have been translated into SQL script.

Business Rule	SQL Script	Table

Each department must have exactly one manager, and one manager can manage one department.	`alter table department`
	`Add constraint`
	`departmentEmpID_fk foreign`
	`key(emp_ID) references`
	`employee(Id) on delete cascade`
	`;`
	`create table department (`
	`emp_Id number(10) not null,`
		Department.

Each department must have employee working within it and each employee must belong to at least one department.	`create table works_for (`
	`emp_Id number(10) not`
	`null,`
	`dep_Id number(10) not`
	`null,`
	`constraint worksFor_pk primary`
	`key (emp_Id, dep_Id) )`
	`;`
		works_for

Each member has a membership.	`create table`
	`MemberHasMemberShip (`
	`membership_Id number(10)`
	`not null,`
	`member_ssn number(10) not`
	`null,`
	`membership_Date date,`
	`constraint has_pk primary key`
	`(membership_Id, member_ssn) )`
	`;`
		MemberHasMemberShip

Each class type can be assigned to many classes.	`create table classes (`
	`reference_No number(10)`
	`not null,`
	`Name varchar2(25),`
	`classesTypeId number(10)`
	`not null,`
	`Room_No number(5),`
	`members_Number number(5),`
	`emp_Id number(10),`
	`constraint classes_pk primary`
	`key (reference_No) )`
	`;`
	
	`alter table classes`
	`Add constraint classesEmpID_fk`
	`foreign key(emp_Id) references`
	`Trainer(emp_Id) on delete`
	`cascade`
	`;`
		Classes

Each member has availability to join any class and classes are conducted when there are several members enrolled.	`create table joinClass (`
	`class_Reference_No`
	`number(10) not null,`
	`member_SSN number(10) not`
	`null,`
	`constraint join_pk primary key`
	`(class_Reference_No,`
	`member_SSN) )`
	`;`
		joinClass

Each trainer can conduct classes, and not all trainers should conduct a class.	create table classes (reference_No number(10) not null, Name varchar2(25), classesTypeId number(10) not null, Room_No number(5), members_Number number(5), emp_Id number(10) not null, constraint classes_pk primary key (reference_No)) ; alter table classes Add constraint classesEmpID_fk foreign key(emp_Id) references Trainer(emp_Id) on delete cascade ;	classes
Each trainer can work as a personal trainer for many members in any room.	create table personal_Training (Mem_ssn number(10) not null, roomNo number(5) not null, emp_ID number(10) not null, training_Date varchar2(25), training_Time varchar(10), Extra_Fees number(5), constraint training_pk primary key (Mem_ssn, roomNo, emp_ID)) ;	personal_Training
Each room can be available for a trainer to train many members.	create table personal_Training (Mem_ssn number(10) not null, roomNo number(5) not null, emp_ID number(10) not null, training_Date varchar2(25), training_Time varchar(10), Extra_Fees number(5), constraint training_pk primary key (Mem_ssn, roomNo, emp_ID)) ;	personal_Training
Each member has a trainer in any room.	create table personal_Training (Mem_ssn number(10) not null, roomNo number(5) not null, emp_ID number(10) not null, training_Date varchar2(25), training_Time varchar(10), Extra_Fees number(5), constraint training_pk primary key (Mem_ssn, roomNo, emp_ID)) ;	personal_Training

Each membership can have a discount applied on it.	alter table Membership Add constraint membershipDiscountNo_fk foreign key(discount_No) references discount(discount_No) on delete cascade ;	Membership
There are classes need to use equipment.	alter table Machine_And_Equipment Add constraint machineClassRefNo_fk foreign key(class_Reference_No) references classes(reference_No) on delete cascade ;	Machine_And_Equipment
Each member can have many In-body test.	alter table InBody_Test Add constraint InBody_Test_fk foreign key(mem_SSN) references Members(ssn) on delete cascade ;	Inbody_test
Each room can hold many classes, and each class take a place in one room.	alter table classes Add constraint classesRoomNo_fk foreign key(Room_No) references rooms(roomNo) on delete cascade ;	Classes

8 Queries

In the following subsections, there are five different SQL queries which implements five of the indented output of our system.

8.1 *<Male Members>*

Query in natural language (ENGLISH)

Display id and first name of all male members in ascending order.

SQL script

```
select members.id, fname
from members
where sex = 'M'
order by members.id asc;
```

Caption of the first five rows of the output

ID	FNAME
1907117617	Hyatt
1977382735	Kamal
2005431758	Brent
2017528465	Gannon
2083841816	Farrah
2123268100	Tashya

8.2 *<personal trainer older than 35>*

Query in natural language (ENGLISH)

Display all the names personal trainer that are older than 35.

SQL script

```
select fname ,mname
from  employee
where Id IN (select emp_ID from Personal_Training) and age >35
```

Caption of the first five rows of the output

```
select fname ,mname
from  employee
where Id IN (select emp_ID from Personal_Training) and age >35
```

FNAME	MNAME
Xenos	Kevin
Ross	Samson
Mollie	Forrest

Download CSV

3 rows selected.

8.3 *<Personal Training Members>*

Query in natural language (ENGLISH)

Display all the ids of members that are enrolled in personal training.

SQL script

```
select Mem_ssn
from personal_training
where mem_ssn in  (select ssn  from members);
```

Caption of the first five rows of the output

MEM_SSN
1073351588
1074485447
1128662007
1136646606
1145628839

8.4 < Black Friday *Discount* >

Query in natural language (ENGLISH)

Display the number and amount of all black Friday discount

SQL script

```
select discount.discount_No , discount.discount_Amount
from discount
where offer_Name  = 'Black Friday';
```

Caption of the first five rows of the output

DISCOUNT_NO	DISCOUNT_AMOUNT
7862005	40

8.5 *<Year of Experience>*

Query in natural language (ENGLISH)

Display the year of experience and name for every trainer.

SQL script

```
select trainer.Year_of_Experience, employee.fname
from trainer, employee
where trainer.emp_Id   = employee.id;
```

Caption of the first five rows of the output

YEAR_OF_EXPERIENCE	FNAME
6	Rafael
5	Mollie
1	Daria
9	Xenos
4	Ross

APPENDIX

Members Table:

SSN	ID	FNAME	MNAME	LNAME	EMAIL	CITY	STREET	BUILDING	SEX	BIRTH_DATE	PHONE_NO
1178115405	1906317781	Bevis	Rudyard	Bert	feugiat@yahoo.edu	Berlin	1973	38	F	07-MAY-95	564884477
1126832224	2179332788	Ishmael	Cain	Cairo	ridi@hotmail.com	Berlin	1993	33	F	19-AUG-90	585153391
1144307545	2182231477	Derek	Cade	Abel	amet@outlook.com	Berlin	6316	45	F	19-SEP-88	516674449
1145628839	2022734363	Rashad	Josiah	Shad	nisi@yahoo.org	Berlin	6357	11	F	30-SEP-01	541315532
1128662007	1956775779	Mariko	Honorato	Carl	infg@yahoo.org	Berlin	1098	77	F	02-APR-92	500531332
1136646606	2005431758	Brent	Ivan	Cody	commodo@aol.couk	Berlin	4417	68	M	05-JAN-02	577235214
1074485447	2151284586	Bert	Damian	Damian	phar@yahoo.org	Berlin	7496	82	F	07-MAR-98	586548202
1073351588	2017528465	Gannon	Kieran	Martin	aliqu@google.ca	Berlin	3828	13	M	29-MAR-86	548409381
1163333028	2176844559	Kyra	Cyrus	Harlan	sed@outlook.org	Berlin	8562	13	F	15-OCT-00	586457816
1125698781	2039305666	Sierra	Blaze	Jakeem	lectus@aol.edu	Berlin	8577	3	F	04-JAN-96	544745349
1167615372	1977382735	Kamal	Rogan	Carson	in.mi@gmail.com	Berlin	4039	6	M	01-JAN-96	532730811
1002640405	2083841816	Farrah	Chaney	Morse	aenean@outlook.com	Berlin	2872	32	M	09-MAY-83	555535857
1104445521	1907117617	Hyatt	Elton	Pugh	sit.amet@gmail.com	Berlin	2363	95	M	09-SEP-90	551494745
1180637464	2043537306	Kaye	Alden	Calhoun	luctus.aliquet@gmail.com	Berlin	9671	22	F	07-JUN-66	566816172
1040559581	1956826453	Francis	Blaze	Olsen	non.enim@gmail.com	Berlin	9045	7	F	03-MAY-81	533199370
1145152146	2123268100	Tashya	Merritt	Morse	dolor.quisque@hotmail.com	Berlin	1740	10	M	07-APR-72	567091192
1025341125	2043320484	Vernon	Duncan	Mcgowan	nisi@hotmail.com	Berlin	2166	82	F	10-APR-68	521224113
1016322546	2172186736	Indira	Hyatt	Garrett	est@hotmail.com	Berlin	1765	77	F	01-SEP-99	544766689
1136423644	1936006315	Yuri	Baker	Hodges	nunc@gmail.com	Berlin	9156	15	F	06-JAN-82	516448843

Trainer

EMP_ID	QUALIFICATION	YEAR_OF_EXPERIENCE
1846413440	Level 1 Certificate in Fitness Instructing	1
1845459625	Level 3 Diploma in Instructing Exercise and Fitness	6
1875554684	Level 2 Diploma in Fitness Instructing	5
2208777075	Level 3 Certificate in Fitness Instructing	9
1808547487	Level 1 Certificate in Fitness Instructing	4

InBody_Test

ID	AGE	HEIGHT	WEIGHT	TEST_DATE	MEM_SSN
7	24	170	94	22-DEC-21	1178115405
2	49	174	86	18-JAN-20	1073351588
3	22	173	49	09-APR-20	1074485447
5	22	179	64	26-MAY-21	1136646606
6	48	163	46	10-JUN-21	1128662007
4	24	166	47	21-NOV-20	1128662007
1	45	182	77	09-JAN-20	1145628839

department

ID	NAME	BUILDING	STREET	CITY	EMP_ID
1	Marcia	1	3360	Berlin	1933695381
2	Donna	1	3360	Berlin	1989841691
3	Jeanette	1	3360	Berlin	1990801396
4	Ane	1	3360	Berlin	1949384146
5	Ginger	1	3360	Berlin	1847926630
6	Madeline	1	3360	Berlin	2023937661

Employee

ID	SSN	FNAME	MNAME	LNAME	AGE	JOP_TITLE	PHONE_NO
2205421633	1053524391	Brent	Ivan	Cody	20	Front desk associate	548737117
1863471223	1121263034	Bert	Damian	Damian	42	Nutrition consultant	533062253
1922688762	1026580752	Penelope	Beck	Kibo	23	Front desk manager	585581556
1986973764	1061132118	Leilani	Ronan	Kyle	25	Wellness coach	585676661
1956728590	1137249963	Gannon	Kieran	Martin	34	Members coordinator	573529354
1985521633	1052766645	Darryl	Talon	Stewart	20	Fitness consultant	558827572
1845459625	1186297024	Rafael	Tyler	Allen	25	Trainer	503341206
1875554684	1081515853	Mollie	Forrest	Jameson	50	Trainer	528162110
1846413440	1122713593	Daria	Magee	Samuel	40	Trainer	546368149
2208777075	1160522784	Xenos	Kevin	Garrison	42	Trainer	586671623

2208777075	1160522784	Xenos	Kevin	Garrison	42	Trainer	586671623
1933695381	104735651	Ayanna	Zachery	Myles	46	Manager	545238151
2023937661	112942719	Quinlan	Drew	Alexander	35	Manager	525598889
1989841691	115320877	Hyatt	Jamal	Devin	39	Manager	576981347
1990801396	100088001	Camilla	Julian	Quinn	49	Manager	547326333
1949384146	116217220	Buffy	Lewis	Graiden	49	Manager	553255582
1847926630	100480087	Ethan	Colton	Upton	40	Manager	552536328
1808547487	116697333	Ross	Samson	Ignatius	45	Trainer	518825955
2162677935	116259863	Calista	Lester	Dominic	36	Trainer	529843894

Membership

ID	NAME	DURATION	PAYMENT_AMOUNT	DISCOUNT_NO
2975240001	Gold	30	600	7862003
2975240002	Platinium	30	800	7862005
2975240003	Platinium	60	1600	7862002
2975240004	Gold	90	1800	-
2975240005	Diamond	90	3000	7862005
2975240006	Silver	365	4800	7862002
2975240007	Platinium	365	9600	7862004
2975240008	Gold	30	600	-
2975240009	Diamond	90	3000	-
2975240010	Diamond	30	1000	7862002

2975240011	Platinium	180	4800	-
2975240012	Diamond	60	2000	7862004
2975240013	Silver	180	2400	7862002
2975240014	Gold	60	1200	-
2975240015	Silver	90	1200	-
2975240016	Platinium	90	2400	7862005
2975240017	Diamond	30	1000	7862003
2975240018	Diamond	60	2000	7862002
2975240019	Gold	90	1800	-

Classes

REFERENCE_NO	NAME	CLASSESTYPEID	ROOM_NO	MEMBERS_NUMBER	EMP_ID
26001	HIIT Class	285002	1	3	1875554684
26002	Yoga Class	285006	202	10	1846413440
26003	Boxing Class	285005	203	4	1846413440
26004	Cardio Class	285004	103	5	1875554684
26005	Strength Training Class	285001	101	6	1845459625
26006	Water Aerobics Class	285003	2	7	2208777075
26007	Zumba Class	285007	2	7	2208777075

Machine_And_Equipment

ID	NAME	CLASS_REFERENCE_NO
15000	Smith Machine	26001
15001	Smith Machine	26001
15002	Biceps	26001
15003	Yoga Mat	26002
15004	Yoga Mat	26002
15005	Yoga Mat	26002
15006	Yoga Mat	26002
15007	Yoga Mat	26002
15008	Yoga Mat	26002
15009	Yoga Mat	26002
15010	Yoga Mat	26002
15011	Yoga Mat	26002
15012	Yoga Mat	26002
15013	Boxing Gloves	26003
15014	Boxing Gloves	26003
15015	Boxing Gloves	26003
15016	Boxing Gloves	26003
15017	Boxing Hand Wraps	26003
15018	Boxing Hand Wraps	26003
15019	Boxing Hand Wraps	26003
15020	Boxing Hand Wraps	26003
15021	Punching Bag	26003

15022	Punching Bag	26003
15023	Punching Bag	26003
15024	Punching Bag	26003
15025	Indoor Cycling	26004
15026	Indoor Cycling	26004
15027	Indoor Cycling	26004
15028	Stair Climber	26004
15029	Stair Climber	26004
15030	Leg Press	26005
15031	Leg Press	26005
15032	Lat pulldown	26005
15033	Lat pulldown	26005
15034	Chest Press	26005
15035	Chest Press	26005

class_Type

ID	NAME	DESCRIPTION
285001	Strength Training Class	In a strength training class, you'll be moving heavy objects, or your body, in a way that creates resistance.
285002	HIIT Class	High-intensity interval training (HIIT) workouts are a popular kind of exercise that involve exerting your maximum energy during short periods of time
285003	Water Aerobics Class	A low impact but high intensity water aerobics class where no swimming skills are needed.
285004	Cardio Class	Cardio Class provide Type of exercise that gets your heart rate up and keeps it up for a prolonged period of time.
285005	Boxing Class	Boxing class teaches the basic boxing stance, basic footwork, and how to throw a punch while also gradually building up your stamina, strength.
285006	Yoga Class	Yoga class incorporates yoga postures, gentle movement sequences, breathwork, supported silent meditation, and guided relaxation.
285007	Zumba Class	Zomba dance class features high- and low-intensity intervals that help improve cardiovascular fitness while also enhancing balance, coordination, agility.

Discount

DISCOUNT_NO	DISCOUNT_DATE	OFFER_NAME	DISCOUNT_AMOUNT
7862002	01-JAN-23	New Year	15
7862003	01-JUN-23	Summer Season	5
7862004	04-JUL-23	National Day	20
7862005	25-NOV-23	Black Friday	40

Rooms

ROOMNO	FLOOR
1	G
2	G
3	G
101	F
102	F
103	F
201	S
202	S
203	S

personal_Training

MEM_SSN	ROOMNO	EMP_ID	TRAINING_DATE	TRAINING_TIME	EXTRA_FEES
1145628839	102	1845459625	Sunday, Tuesday, Thrusday	3:00 PM	300
1073351588	3	2208777075	Monday, Wednesday	6:30 PM	200
1074485447	203	1808547487	Sunday, Tuesday, Thrusday	8:00 PM	300
1128662007	101	1808547487	Sunday, Tuesday, Thrusday	1:00 PM	300
1136646606	202	1875554684	Monday, Wednesday	6:00 PM	200

works_for

EMP_ID	DEP_ID
1808547487	3
1808547487	6
1845459625	2
1846413440	2
1846413440	4
1846413440	5
1863471223	1
1863471223	5
1875554684	3
1875554684	6

1922688762	2
1956728590	3
1985521633	6
1986973764	4
2162677935	1
2205421633	5
2208777075	4

MemberHasMemberShip

MEMBER_SSN	MEMBERSHIP_ID	MEMBERSHIP_DATE
1178115405	2975240007	04-JUL-22
1126832224	2975240006	01-JAN-23
1144307545	2975240008	11-FEB-23
1145628839	2975240004	21-JAN-23
1128662007	2975240010	01-JAN-23
1136646606	2975240002	25-NOV-22
1074485447	2975240001	01-JUN-22
1073351588	2975240005	25-NOV-22
1163333028	2975240009	12-DEC-22
1125698781	2975240003	01-JAN-23
1145152146	2975240011	04-OCT-22
1167615372	2975240012	04-JUL-22
1016322546	2975240013	01-JAN-23
1002640405	2975240014	25-DEC-22
1136423644	2975240015	04-JAN-23
1025341125	2975240016	25-NOV-22
1040559581	2975240017	04-JUL-22
1104445521	2975240018	01-JAN-23
1180637464	2975240019	16-JAN-22

joinClass

CLASS_REFERENCE_NO	MEMBER_SSN
26001	1016322546
26001	1025341125
26001	1136423644
26002	1002640405
26002	1016322546
26002	1025341125
26002	1073351588
26002	1104445521
26002	1126832224
26002	1136423644
26002	1136646606
26002	1167615372
26002	1180637464
26003	1040559581
26003	1104445521
26003	1125698781
26003	1145152146
26004	1025341125
26004	1040559581
26004	1104445521
26004	1145152146
26004	1180637464

26005	1016322546
26005	1040559581
26005	1136423644
26005	1136646606
26005	1144307545
26005	1180637464
26006	1074485447
26006	1126832224
26006	1128662007
26006	1136646606
26006	1144307545
26006	1145628839
26006	1178115405
26007	1074485447
26007	1126832224
26007	1128662007
26007	1136646606
26007	1145628839
26007	1163333028
26007	1178115405